CSCI 2321 (Computer Design), Spring 2021

Homework 7

Credit: 20 points.

1 Reading

Be sure you have read, or at least skimmed, the assigned sections of Chapter 4 of the text, starting with 4.6.

2 Problems

Answer the following questions. You may write out your answers by hand and scan them, or you may use a word processor or other program, but please submit a PDF or plain text via e-mail to my TMail address. (No links to shared files on Google Drive please.) Please use a subject line that mentions the course and the assignment (e.g., "csci 2321 hw 7" or "computer design hw 7").

1. (15 points) We have looked at two implementations for a subset of the MIPS instruction set architecture, single-cycle and pipelined. For each of the following statements, say whether it is true or false and why. (*Hint:* None of these are intended to be trick questions.)

   (a) The cycle time for the pipelined implementation will probably be shorter than for the single-cycle implementation. (Recall from Chapter 1 that cycle time and time to execute an instruction might or might not be the same.)

   (b) All instructions take the same amount of time in the single-cycle implementation (counting from when the instruction begins to when the next instruction begins).

   (c) All instructions take the same amount of time in the pipelined implementation (meaning from the time the instruction is fetched from memory to the time it completes), *if control and data hazards are ignored*.

   (d) All instructions take the same amount of time in the pipelined implementation (meaning from the time the instruction is fetched from memory to the time it completes), *if control and data hazards are NOT ignored*.

   (e) Most programs will probably run faster on the pipelined implementation.

   (f) Aside from possible differences in execution time, programs will run the same way (in the sense of giving the same results) on both implementations. (You can assume that the hardware correctly resolves any control and data hazards — if it doesn't, it's broken!)

2. (5 points) How many cycles are needed to complete execution of a program with a million instructions if using a pipelined implementation with five pipeline stages? If it depends, on what, and what's the maximum and minimum? (For the maximum, explain how you came up with the number you did; on reflection I realize that I'd have a tough time being sure about this myself!)

3 Pledge

For programming assignments, this section should go in the body of the e-mail or in a plain-text file `pledge.txt` (no word-processor files please). For written assignments, please put it in the text or PDF file with your answers.

Include the Honor Code pledge or just the word "pledged", *plus* at least one of the following about collaboration and help (as many as apply). Text *in italics* is explanatory or something for you to fill in.

- I did not get outside help *aside from course materials, including starter code, readings, sample programs, the instructor.*
- I worked with *names of other students* on this assignment.
- I got help with this assignment from *source of help — ACM tutoring, another student in the course, etc. (Here, "help" means significant help, beyond a little assistance with tools or compiler errors.)*
- I got help from *outside source — a book other than the textbook (give title and author), a Web site (give its URL), etc.. (Here too, you only need to mention significant help — you don't need to tell me that you looked up an error message on the Web, but if you found an algorithm or a code sketch, tell me about that.)*
- I provided help to *names of students* on this assignment. *(And here too, you only need to tell me about significant help.)*

4 Essay

For programming assignments, this section should go in the body of the e-mail or in a plain-text file `pledge.txt` (no word-processor files please). For written assignments, please put it in the text or PDF file with your answers.

Include a brief essay (a sentence or two is fine, though you can write as much as you like) telling me what if anything you think you learned from the assignment, and what if anything you found interesting, difficult, or otherwise noteworthy.